

IN THE CLAIMS:

Please cancel Claims 9 to 12, 16, 27 and 32 without prejudice or disclaimer of subject matter, and amend Claims 1, 3, 6 to 8, 14, 17 to 22, 24, 28 and 33 as shown below. The claims, as pending in the subject application, now read as follows:

1. (Currently amended) A computer implemented method for generating a device driver for an output device by an information processing apparatus, wherein said output device is connected to said information processing apparatus, said method comprising the steps of:

loading an application from a read-only memory, said application including a device model independent device driver;

determining a model of said output device to which said application is intending to issue output commands;

determining whether a device model dependent configuration data in a memory device matches said model of said output device, said memory device being a memory card connected to said information processing apparatus;

upon determining that said device model dependent configuration data in said memory device matches said model of said output device, reading said device model dependent configuration data from said memory device; and

generating said device driver for said output device by configuring said device model independent device driver with said device model dependent configuration data.

2. (Canceled)

3. (Currently amended) A method according to claim 1, wherein said device model dependent configuration data is color conversion data to convert RGB color-space to a native color-space of said output device.

4. and 5. (Canceled)

6. (Previously presented) A method according to claim 1, wherein said application in said read-only memory is an unchangeable application.

7. (Original) A method according to claim 1, wherein said information processing apparatus is a game console.

8. (Previously presented) A method according to claim 1, wherein said application is a game application executed on a game console.

9. to 13. (Canceled)

14. (Currently amended) A method according to claim 1, wherein said model of said output device is determined through reading an identification string from said output device.

15. and 16. (Canceled)

17. (Previously presented) A method according to claim 1, wherein, upon determining that said device model dependent configuration data in said memory device does not match said model of said output device, said determining step further determines whether further configuration data in a further memory device matches said model of said output device, and upon determining that said further configuration data in said further memory device matches said model of said output device, loading said further configuration data from said further memory device as said device model dependent configuration.

18. (Previously presented) A method according to claim 1, wherein said read-only memory is an optical storage disc for a game.

19. (Original) A method according to claim 1, wherein said device is connected to said information processing apparatus by means of a USB connection.

20. (Previously presented) A method according to claim 1, wherein said output device is a printer, and said loading configuration step is executed in the information processing apparatus when said application is controlled to perform a printing operation.

21. (Currently amended) An information processing apparatus for executing an application and for generating a device driver for an output device connected to said information processing apparatus, said information processing apparatus comprising:

a read-only memory for storing an application, said application including a device model independent device driver;

a memory device for storing device model dependent configuration data, said memory device being a memory card connected to said information processing apparatus; and

a processor for determining a model of said output device to which said application intends to issue output commands, determining whether said device model dependent configuration data matches said model of said output device, upon determining that said device model dependent configuration data in said memory device matches said model of said output device, reading said device model dependent configuration data from said memory device and generating said device driver for said device by configuring said device model independent device driver with said device model dependent configuration data.

22. (Currently amended) A computer program to be executed in an information processing apparatus for executing an application and for generating a device driver for an output device connected to said information processing apparatus, said computer program being stored on a read-only memory and comprising:

a device model independent device driver;

code for determining a model of an output device to which said application intends to issue output commands;

code for determining whether device model dependent configuration data in a memory device matches said model of said output device, said memory device being a memory card connected to said information processing apparatus;

upon determining that said device model dependent configuration data in said memory device matches said model, code for reading said device model dependent configuration data from said memory device; and

code for generating said device driver for said output device by configuring said device model independent device driver with said device model dependent configuration data.

23. (Canceled)

24. (Currently amended) A method of providing forward compatibility of device driver code of an unchangeable application with a plurality of device models, wherein said application is stored in a read-only memory and not linked to other executable code, said method comprising the steps of:

including device model independent device driver code in said application;

determining a model of an output device which said application is desired to issue commands to;

reading model dependent configuration data for said model of said output device from a memory card; and

generating a device driver for said model of said output device by configuring said device model independent device driver code with said model dependent configuration data.

25. (Original) A method according to claim 24, wherein said model of said device is determined through reading an identification string from said device.

26. (Original) A method according to claim 24 or 25, wherein said unchangeable application is a game application executed on a game console.

27. (Canceled)

28. (Currently amended) An information processing apparatus for providing forward compatibility of device driver code of an unchangeable application with a plurality of device models, wherein said application is stored in a read-only memory and not linked to other executable code and said device driver code is device model independent, said apparatus comprising:

a memory card storage means for storing model dependent configuration data for a plurality of devices;

means for determining a model of an output device which said application is desired to issue commands to;

data reading means for reading model dependent configuration data associated with said model of said output device from said memory card storage means; and

means for generating a device driver for said model of said output device by configuring said device model independent device driver code with said model dependent configuration data.

29. (Previously presented) Apparatus according to claim 28, wherein said output device is connected to said apparatus by means of a USB connection.

30. (Previously presented) Apparatus according to claim 28 or 29, wherein said model of said output device is determined through reading an identification string from said output device.

31. (Previously presented) Apparatus according to any one of claims 28 to 30, wherein said unchangeable application is a game application and said apparatus is a game console.

32. (Original) Apparatus according to claim 31, wherein said storage means is a memory card of said game console.

33. (Currently amended) A computer program product, carried on a read-only storage medium, for providing forward compatibility with a plurality of device models, wherein said computer program product is not linked to other executable code, said computer program product comprising:

an application;

code for determining a model of an output device which said application is desired to issue commands to;

code for reading model dependent configuration data associated with said model of said output device from a memory card ~~device~~; and

code for generating a device driver for said model of said output device by configuring device model independent device driver code with said model dependent configuration data.